User Preference Modeling by Global and Individual Weights for Personalized Recommendation

Ondrej Kassak, Michal Kompan, Maria Bielikova

Slovak University of Technology in Bratislava, Faculty of Informatics and
Information Technologies, Ilkovičova 2, 842 16 Bratislava 4, Slovakia
{name.surname}@stuba.sk

Abstract: In this paper we propose a novel user model for personalized recommendation in domains, where items are described by multiple characteristics (e.g., metadata attributes) and users' preferences are expressed on the level of items by some kind of explicit feedback (e.g., rating) or derived from implicit user feedback (e.g., time spent on items). The proposed user model is composed of two parts. The first one represents user preferences by items descriptive metadata (e.g., genres, keywords, actors and directors in movie domain used as an example in this paper) expressed as preference vectors. The second part is represented by a vector describing user's preferences on previously rated items. Our main contribution is the addition of individual user preference measures to the vectors of both user model parts. User preferences expressed as the vector weights are calculated based on the global preference of large amount of users and also on individual variations in preferences of particular users to the average of the whole set of users. The weights reflect how much the different types of information are important to the user. We evaluated the proposed user model in movie domain through collaborative recommendation by comparing its performance to two reference user models using Movielens 10 M dataset.

Keywords: user model; weighted vector; user similarity; personalized recommendation; multimedia domain; movie recommendation

1   Introduction and Related Work

Information overload problem over the Web is often reduced by personalized recommendation. Methods of personalized recommendation are generally based on a user model that estimates the user's preferences or his/her interests relating to specific domain.

The issue of reducing information overload is currently researched for a number of domains, such as multimedia, texts, jobs, etc. From the user modeling point of view, there are several similar characteristics typical for these domains. At first,

these domain items are described by various metadata attributes. These attributes can be derived either from the items' content (e.g., keywords, header, subject) or can describe some items' characteristics (e.g., author, timestamp, genre, popularity, or sentiment [4]). In this way each item is characterized by a set of information. Moreover, the knowledge of user preferences is crucial to be able to model his interests. The feedback is captured in various forms from explicit ratings to implicit visits, time spent on items by the user, clicks or other signals of the user activity [9].

In this paper we propose a novel approach for user modeling for personalized recommendation in domains, where items are described by multiple characteristics and users' preferences are expressed on the level of items. We focus on the movie domain as a typical representative of above described domain characteristics, where is a huge users' activity over various web-based systems and services (expanding to more and more devices such as personal computers, TVs, mobile devices, etc.). The average U.S. citizen watches TV daily for 5.11 hours, which results in a total of 9 years during the person's lifetime. A schoolchild in the U.S. watches television an average of 1 200 hours per year, while they only spend 900 hours in the school classroom during the same time period (BLS American Time Use Survey, A.C. Nielsen Co., 2012). These statistics cover just watching TV, they do not indicate watching videos and movies on the internet, so we can expect these numbers to be actually even higher. Therefore, it is appropriate to help users in finding various options to choose items that interest them.

Humans in general have different interests and attribute different importance to things in comparison to other people [13]. For example, somebody chooses movies primarily by genre, another one by its director. At the same time, domain of movies is quite well-structured, i.e. the items like movies can be described by metadata that form several categories (e.g., genres, directors). Hu et al. in their work report that users tend to rate similarly the items which have multiple metadata elements in common [14]. This information in addition to the suitable metadata structure allows us to model user preferences by the mean of overlay user models and thus reflects user preferences in more detail. Our approach to user modeling follows the overlay user model concept. We proposed a mechanism which dynamically adjusts general weights according to individual user's behavior and the whole group of users' activities.

In the most general view Senot et al. [21] defined the user model, as a set of pairs [concept, value], while the value is from the interval <0; 1>, which represent a user's level of interest to the category of items (e.g., movies directed by Quentin Tarantino). The rate of interest is determined for example as the average amount of user's ratings of items in this category. Except the preference rate, Wang et al. propose to differ between long and short term preferences, which express the stability of user interest [23]. This could be realized for example by introducing also the concept of weights used to store the number of concept ratings.

The most widespread approach to user modeling (besides the Bayesian networks [7] and graph application [2, 22]) models user preferences based on individual users' preferences [5]. In movie domain such user models are often represented by a set of vectors. The vector models formed by genres, actors and keywords represent the basis for collaborative recommendation according to Yu et al. [26]. The authors in [12] use the genres as the only source of information. Mukherjee et al. defines user model as a set of genres, actors, actresses and director's vectors [19]. Wang et al. extends this concept to a set of multiple vectors, where each of them is used for different user's mood [24], which can be automatically recognized [17]. Vector user model formed by actors, directors, genres and keywords, where each vector element has calculated its own static weight, is used also by Debnah et al. [10]. Mutual similarities between the vector users models are generally computed using metrics such as Cosine similarity, Pearson correlation, Euclidean distance, etc. [24].

The main problem is that existing user models, do not sufficiently model what users seen as important and according to what aspects they decide primarily when selecting an item. This is possible to solve by understanding how much users are alike, not only by the selected items, but also according to the method of choice (decision making process). This way we improve the information value provided by the user model by the mean of more specificity levels, which is then used for recommendation. A similar idea is actually researched in the movie domain by Wen and Chen, who except for user ratings aim to find user similarity measure primarily by their personality traits, based on idea that users choose firstly by their personalities [25].

Our proposed approach estimates user's preferences not only from the low level preferences (e.g., specific genre or director) perspective, but as well, we model the higher level – found patterns based on how the user makes decisions (e.g., user attribute the importance to genres, or directors). In other words, not only the preference of specific genres (e.g., comedy) directors (e.g., Tarantino) is reflected in the proposed user model, but the way the user decides (e.g., he/she mainly decides by a director, fewer by genres) is considered primarily. This aspect, nowadays, is not considered in current user modeling approaches in the movie domain.

Our aim is to introduce in this paper a novel user model, which

– considers not only specific preference value for vector items (e.g., preference of comedy genre, Tarantino (director)), but a higher level of user preferences.

Identification of the way how the user decides when he/she chooses a movie seems to be very helpful knowledge for personalized recommendations. This knowledge (e.g., first he/she chooses the genre and after that some director known to him), offers us an advantage in comparison to existing methods of user modeling.

Our proposed concept can be used within several existing approaches as for example in shared interoperable user models [8] where it could be helpful to spread information about user decision preferences. At the same time, it can improve open editable user models [3], where it can suggest model vectors' preferences automatically. Similarly, it could ease the cold start [20] problem by setting the initial preferences.

2   User Model Components

To create a user model or to recommend items in a particular domain, it is necessary to be able to describe distinctive items or users' properties. In the movie domain it is common to get the user's feedback in the form of explicit ratings of items (e.g., movies). Working with multimedia content is, therefore, more challenging due to the lack of automatized approaches for content analysis in such a domain. On the other hand, thanks to this shortcoming, users are often involved by providing detailed information in order to produce metadata describing items characteristics. This rich metadata information such as title, genres, release date, languages, a list of directors, actors, keywords, etc. (typical for well-structured domains) is commonly used in order to model user preferences and to analyze distinctive features of the recommended items.

In order to model user preferences in various levels of abstraction we propose a novel weighted vector user model consisting of a set of vectors, which represent the user's preferences to particular characteristics of multimedia content based on available metadata (e.g., genres, keywords, directors and actors for movies) and the preferences to specific items themselves.

Un-weighted or only statically weighted vector models are currently widely used in various domains. Our contribution consists of the addition of individual weights to vectors forming the user model, while these are dynamically recalculated based on the user's behavior and behavior of all other users, as well. Thanks to our enhancement of individual weights to the model vectors we improved the representation and modeling of user's interests. In our proposed user model, the final weight for every vector is determined as a combination of initial global and individual weight parts.

Initial global weight represents the importance of the vector compared to the other user model's vectors (this helps us express user preferences in the mean of the preference of genres or directors, etc.). Its weight is the same for all users and is used as an initial setting for new users. Opposing this, *individual user weight* reflects variation in the interests of modeled user behavior compared to the average behavior of other users. Vector weights, model the users' decision making process (from the high level preference point of view).

Vectors and their weights in our user model are useful in the step of similar user search (often performed by recommender systems [16]) as it is beneficial to find similar users not only based on the low level user perspective (a user likes some actor), but from the high level perspective as well (user often decides based on actors).

Let D be an assembly of sets of descriptive metadata associated with items from modeled domain and P a set of user's item preferences. The user model UM is then defined as a couple:

$$UM = (D, P) \tag{1}$$

Each element of D represents a set of weighted vectors wv (descriptive metadata vectors as, e.g., genres, keywords, actors and directors for movie domain that aggregate metadata for each category and its values) from the set of all weighted vectors WV (all item metadata types).

$$D = \{wv_1, wv_2, \ldots, wv_{n, n \in \mathbb{N}}\}, wv_{i, i \leq n} \in WV \tag{2}$$

P is a set of weighted user's item preferences (e.g., item ratings for movie domain), which is defined as the weighted vector wv from WV

$$P = \{wv_{m \in \mathbb{N}}\}, wv_m \in WV \tag{3}$$

The weighted vector wv is defined as a couple of a vector v and a final weight w of the vector v

$$wv_i = (v_i, w_i), i \in \mathbb{N} \tag{4}$$

where v is defined as set of k vector elements ve (actual metadata elements, e.g., comedy as actual genre, or Tarantino as actual actor):

$$v = \{ve_1, ve_2, \ldots, ve_{k=|v|}\} \tag{5}$$

The Na is a set of all element names, Pm is an interval of all users' preference measures defined as $<min\ preference;\ max\ preference>$ (e.g., set of possible rating values) and ew a number of elements preferences considered (e.g. number of ratings of the element). Then, the vector element ve is defined as a triple:

$$ve_i = (na_i, v_i, ew_i), i \leq |v|, na_i \in Na, v_i \in Pm, ew_i \in \mathbb{N} \tag{6}$$

The final vector weight w is computed as a sum of initial global weight igw (Equations 8, 9) and individual weight iw (Equation 10) precisely described in the Section 3.

$$w = igw + iw, igw \in \mathbb{R}, iw \in \mathbb{R} \tag{7}$$

In our example from movie domain we consider in the D user model part four weighted vectors wv_i – the genres, keywords, actors and directors. The P is represented by one weighted vector describing the user's preference measure for

items represented by item ratings. Weighted vectors for movie domain are described as follows:

Genres. Movie is usually described by several genres, as it is very rare to describe it by an only single genre. Because of the quality of today's metadata describing the item content, it is impossible to quantify the ratio of importance for several genres assigned to one movie, thus all genres obtain for the one movie an equal importance.

Keywords. In our model, keywords represent important terms, named entities (mainly persons, localities and events) or characteristic activities. Together they create a simple movie characteristic. Keywords are ordered based on their characterization relevancy (or by how many users used them to describe the movie). Thus, we are able to determine the keyword importance from its position in the ordered list assigned to every movie.

Actors. The cast is one of the most important factors, based on users making their decision and judging the movies. Just a presence of some famous actor or actress can be the reason that a user watches a movie, even if it is the only information about this movie he/she has.

Directors. The director is a person who influences the movie character and its atmosphere the most. Some types of users choose movies mostly by directors because they expect some patterns occurring in movies directed by the same director.

In addition to the descriptive attributes mentioned above, which do not capture the specific rated movie, but rather describe their properties (e.g., cast, genre), the user model includes the direct information about user's preferences, which we define for movie domain as user activity history expressed by movie ratings. This allows us to enhance the perception of user's rating history and reflect it in the proposed user model. An example of vectors *v* used in proposed user model for movie domain is shown in Table 1.

Table 1

Example of proposed user model consisting of four descriptive metadata vectors (genres, keywords, directors and actors) and user's items preference vector (rated movies). Vector elements are composed of set of triples element name, value, element weight.

Vector	Vector elements
Genres	[comedy; 4.0; 3], [action; 3.8; 5], [war; 1.5; 2], …
Keywords	[journey; 3.4; 6], [hobbit; 4.5; 3], [shire; 3.1; 1], …
Directors	[jackson; 4.1; 5], [tarantino; 4.2; 3], [lee; 1.2; 2], …
Actors	[parsons; 4.1; 3], [streep; 5.0; 8], [cuoco; 4.1; 3], …
Rated movies	[hulk; 3.0; 1], [avatar; 4.0; 1], [hours, 4.5; 1], …

The proposed user model describes the user's preferences from several different perspectives (e.g., user preference of genres and actors). They are captured in the

form of vectors. The main contribution of the proposed user model is an addition of preference final vector weights *w* to the vectors. The weights are calculated individually for every modeled user. A detailed representation of the user preferences is used mainly to determine what criteria the user selects items by, or to identify the users who have the most similar interests as the actually modeled user. Advantage of the proposed vector model is its extensibility, whereas from the user model perspective vectors represent independent components.

3 Preference Modeling

The proposed user model itself is filled based on user's feedback provided for movie domain in the form of users' explicit items ratings (*movie x user rating*). In our user model we store preferences in the form of the low level characteristics, e.g., genres, actors (see Figure 1 left part). After the user rates an item, the item's metadata elements are adjusted in order to model low level user preferences (see Figure 1 middle and right part). This means that for every metadata element (i.e. for every movie genre, actor, etc.) is its value in the user model updated by actual rating value obtained from the user's feedback. For example, if the user rates a movie, which has defined as metadata 2 genres, 2 keywords, 1 director and 2 actors, there are in total 7 elements added or adjusted in the descriptive metadata part of his/her user model and the movie itself represented by some kind of identifier is added to the preference part of the user model.
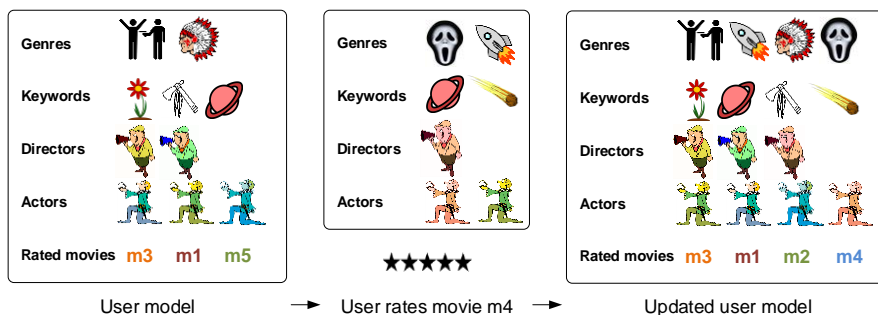

Figure 1

Principle for filling proposed user model by item ratings. The user (represented by the user model - left part, which was created based on three rated movies) gives a rating for a movie, which is described by some metadata (middle part). The rating is added to the user model for each metadata element (e.g., actors, directors, keywords for movie domain). In case the user model already contains some of the metadata elements (e.g., comedy for genres), the element *value* and *weight* are modified for these elements.

If the user rates a new item, vector elements adjustment involves the calculation of element *value* by adding new element *rating* (Equation 8). Consequently, the *weight* of that element is increased (positive rating scale is assumed).

$$value_{i,addition} = \frac{value_i * weight_i + rating_{i,new}}{weigth_i + 1} \tag{8}$$

In a case, the user reevaluates the item he/she has rated in the past, the *values* of all its metadata elements are recalculated just as change of *rating* (Equation 9). The element *weight* does not change in this case because this action means the existing item *rating* is updated, but not added. This ensures the continuous updating of the user model and automatic concept drift handling, which is nowadays actively researched [15].

$$value_{i,change} = \frac{value_i * weight_i - rating_{i,old} + rating_{i,new}}{weight_i} \tag{9}$$

The level of user's interest is derived from the *values* of the individual vectors' elements. Our assumption is that the users are influenced by various *weights* not only in the mean of specific elements (one specific actor, genre, etc.) but also as the entire vectors. Someone chooses movies by actors who play in it or by people who directed it. But, for another user, this information is useless and he/she decides rather by the content (user is interested more in, e.g., keywords).

In other words, proposed approach models user's preferences are not only on the level of specific elements types (one specific genre, actor, etc.), but also the preferences of whole vectors are considered in the decision making process (Figure 2). For each vector the *final* individual preference *vector weight* (blue bar) is obtained by adjusting the *initial global weight* (grey bar, identical for each user) based on the users' decision making preferences – *individual weight* (green and orange bars with sign, different for individual users) – Equation 8. Preference weights can be used in the similarity search phase of recommendation approach.
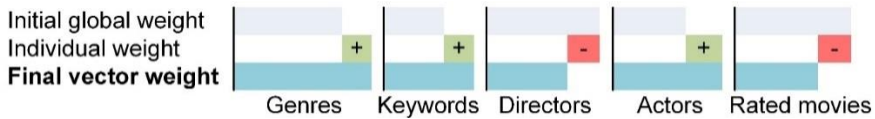

Figure 2

For each vector the final individual preference vector weight (blue bar) is obtained by adjusting the initial global weights (grey bar, identical for each user) by individual weights (green and red bars with sign, different for individual users).

The vector weights for some modeled user are determined as follows:

1) *For each new user assign initial global weights.* This step is the same for each user, weights are determined based on the average user decision making process (average of all users).

2) *Update the initial global weights by individual weights for each user.* Individual weights represent the user's interests' deviation to the average behavior of large amount of users. Thanks to this step, we are able to model user personal preferences on the higher level of abstraction.

Initial global weights. These kinds of weights serve as general rules and this step is obligate for all users equally. Initial global weights are used as starting weight values in the case of a new user (cold start). As user initiates actions (rates items), these weights are changed by the individual weighting mechanism, described below.

Individual weights. Individual preferences are represented as user's preferences deviations in comparison to the average user (computed as the average of preference stereotypes of all users).

Average preferences represent the average interest of all available users in the system. With this kind of interest, we search for each vector in the user model. Individual user preferences of modeled user are computed for each vector as the sum of the deviations to the average preferences divided by the number of vector items (Equation 10). Thus, we get the absolute value of the difference between element frequencies of the modeled user vector and the element frequencies of the average user, called the average deviation (introduced by Ferman et al. [12]). In their work, however, was the sum of deviations calculated as elements division. In our user model we use subtraction because it captures (based on comparison we performed) the real observed deviations more precisely.

$$individual\ weight = \frac{\sum_{i=0}^{N} \left| \overline{item\ weight_i} - item\ weight_i \right|}{N} \tag{10}$$

where *N* is the number of elements of the vector. *Weight* of each element reflects the number of rated items, which metadata contains this element. For example, if we calculate the *weight* of a comedy element belonging to the genres vector, we consider rated movies that contain genre comedy. After computation of the all vector preferences in both model parts, (all metadata vectors and the items vector), the measured values are normalized so the sum of their values will by equal to 1 as:

1) According to the previously described procedure (Equation 10) calculate individual preference of all vectors in a user model.

2) Calculate the inverse sum of absolute values of preferences from step 1.

3) For each vector multiply results of step 1 and step 2.

Normalization process gives for each preferences vector a value from the interval <0; 1>. Preferences represent the personal user model vector importance for modeled user.

4   Experiment Evaluation

We evaluated the proposed user model indirectly by the personalized recommendation based on several reference user models, which is commonly used in domain on user modeling evaluation [18].

Our hypothesis is that the proposed user model including vectors of elements that describe rated items and considering personal preferences of these vectors improves precision of collaborative recommendation in comparison with a user model without mentioned features.

To verify our hypothesis we compared precision of collaborative recommendation based on proposed user model and reference user model generally used in the multimedia domain [1, 11].

4.1   Dataset and Methodology

In order to evaluate proposed weighted vector user model we used Movielens 10 M dataset (http://grouplens.org/datasets/movielens/). The dataset contains ratings from nearly 70 k users (min. 20 ratings per user) on over 10.5 k items. Each rating is characterized by unique identifier, a string containing the name in English followed by release year, set of movies genres and the rating value.

Since the dataset contains only genres of movies as the metadata information, we needed to expand the metadata information. We used the data obtained from the Internet Movie Database (IMDb) including the original item title, the official translation of title for the English language, list of genres, keywords, actors, directors along with information about them, the item content, release date, duration, etc. As an extension of these IMDb data, we modified the set of keywords because of their low quality. We used a text analysis based on the TF-IDF algorithm and replaced some keywords with synonyms and adjusted their weights by re-order.

Process of items mapping was realized by the means of matching pairs of items from IMDb and Movielens (based on the release date and the title). As the result, we obtained over 8.5 million ratings for which we were able to clearly match the rated item and its description from IMDb. In this manner we obtained ratings from 3 k users. Together, processed users give over 331 k ratings which resulted in over 4.5 million of mutual similarities between the pairs of users in each recommendation method used for experiments.

For each user we randomly assigned his/her ratings to the 80% training and 20% testing set. Using the training set, we created user models (for our proposed vector model and reference models respectively). Based on these models, we have calculated the mutual similarity between users and then produced the top-N collaborative recommendations.

4.2   Results

For evaluation of proposed user model we compared it to the reference user model generally used in the multimedia domain [1, 11]. The comparison was realized as collaborative user top-N recommendation. Generated recommendations were based on the reference user model used cosine similarity (as the proposed model did) and respectively the Pearson correlation. For quantitative evaluation we used the standard widely-used precision metric, which represents the proportion of items selected by the user from the recommended set and the total number of recommended items available (prec@1, 3, 5, 10, 15).

Results clearly show (Table 2) that the top-N collaborative recommendation based on the proposed user model achieves statistically significant better results than compared recommendation based on reference user model in the movie domain using cosine similarity and the Person correlation. When recommending one item (prec@1) our method achieves the precision of 61.61%, what means an improvement of 17.20% comparing to the cosine similarity and 23.04% comparing to the Pearson correlation. When recommending the list of 15 items (prec@15) our method achieved precision of 42.94% and improvements of 7.98% and 11.26%.

Table 2
Results of the experiment comparing the precision of collaborative recommendation based on various approaches of the users similarity computation step (CS – cosine similarity, PC – Pearson correlation)

	Weighted vector UM	Reference UM using CS [1, 11]	Reference UM using PC [1, 11]
prec@1	61.61	44.41	38.57
prec@3	51.53	36.74	34.48
prec@5	48.38	36.10	33.51
prec@10	44.35	35.30	32.45
prec@15	42.94	34.96	31.68

The paired T-test was performed in order to investigate the statistical significance. Obtained results are by the general criteria considered as highly statistically significant ($p = 0.0047, \alpha = 0.05$ and $t = 7.6401$ for cosine similarity and $p = 0.0058, \alpha = 0.05, t = 7.0895$ for the Pearson correlation). From these results it is clear that the top-N recommendation using the proposed user model achieves significantly higher precision in comparison to the same recommendation based on reference models.

As we can see our proposed user model achieves the best results when recommending a lower number of items. In fact, this represents in the domain of movie recommendation the ideal situation. In this domain, it takes relatively long time to experience some item and users usually watch only one movie per session. It is, therefore, unnecessary to recommend him/her too many items, but on the other hand, the precision of first few recommended items is crucial from the users' satisfaction point of view.

Conclusions

The main contribution presented in this paper lays in the proposal of a personalized weighting approach for vectors describing users' preferences, which allows us to model user's preferences from various perspectives. Vector weights reflect the high level importance of the information contained in the vector (e.g., genres, keywords, directors) for the modeled user. The proposed user model consists of two parts – *descriptive metadata part* consisting for movie domain of four vectors (genres, keywords, directors and actors) and the *items preference part* described by one vector (rated items identifiers for movie domain). Vectors are composed of elements that capture user's interests based both on metadata and users' rating activity.

The user model principle is generally reusable to any domain, in which items can be described by metadata. An example is the domain of news [6] where, similarly to movie domain, items are described by a set of metadata attributes and there is a need to consider the vector importance. In general, for the user model to be reused it is necessary only to extract the most descriptive (or distinctive) metadata from the target domain and replace the currently used ones. The vector weight expresses how much is the particular type of information described by the vector (e.g., genre, director) important for the user when he/she is choosing some items. In other words, we model general preferences of user decisions, e.g., whether the user generally decides according to a director or rather by genres. For new users (we do not have enough user rating history), vectors' weights are calculated based on the generally applicable rules – the global vector weights. Next, as the user rates more items, they are adjusted by individual weights. Individual weights express how much the user differs from the average behavior of (possibly a large number of) other users.

The proposed user model can be extended by adding further vectors reflecting specific domain characteristics. This extension is from the computational perspective of proposed approach is not critical because it increases linearly. The process of updating elements to the user model is in fact inexpensive operation consisting of updating values and weights only of elements describing the item. An update operation was, in Section 3, illustrated on the explicit user rating, which is the most common format of user feedback used in the movie domain. Any other user feedback format can be used, which enables us to describe user interests on the level of items (the numeric interval expressing user's interest is required). The only requirement is to have data (measure) on user interest related to the items.

The benefits of the proposed user model are mostly in achieving a higher precision in recommendations based on the proposed user model, compared with the reference user models. The advantage of our user model is also the coverage of the user's interests from multiple perspectives. Moreover, by considering the global and individual weights, the proposed approach is able to help the recommendation algorithms to overcome some problems, such as a cold start. For the new user we

can apply general rules for the decision making preferences and start recommend suitable items as soon as possible.

References

[1]   G. Adomavicius, A. Tuzhilin: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, IEEE Transactions on Knowledge and Data Engineering 17(6), (2005) pp. 734-749

[2]   C. C. Aggarwal, J. L. Wolf, K. L. Wu, P. S. Yu: Horting Hatches an Egg: A New Graph- Theoretic Approach to Collaborative Filtering, Proceedings of the 5th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '99) (1999) pp. 201-212, ACM, New York, NY, USA

[3]   J. Ahn, P. Brusilovsky, S. Han: Personalized Search: Reconsidering the Value of Open User Models, Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI '15) (2015) pp. 202-212, ACM, New York, NY, USA

[4]   O. Appel, F. Chiclana, J. Carter: Main Concepts, State of the Art and Future Research Questions in Sentiment Analysis, Acta Polytechnica Hungarica, 12(3) (2015)

[5]   M. Barla, M. Tvarožek, M. Bieliková: Rule-based User Characteristics Acquisition from Logs with Semantics for Personalized Web-based Systems. In Computing and Informatics, 28(4) (2009)

[6]   M. Bieliková, M. Kompan, D. Zeleník: Dušan, Effective Hierarchical Vector-Based News Representation for Personalized Recommendation. Computer Science and Information Systems 9(1) (2012) pp. 303-322

[7]   J. S. Breese, D. Heckerman, C. Kadie: Empirical Analysis of Predictive Algorithms for Collaborative Filtering, Proceedings of the 14th conference on Uncertainty in artificial intelligence (UAI'98) (1998) pp. 43-52, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA

[8]   M. Chevalier, C. Julien, C. Soulé-Dupuy: User Models for Adaptive Information Retrieval on the Web: Towards an Interoperable and Semantic

Model. International Journal Adaptive, Resilient and Autonomic Systems, 3, 3 (July 2012) pp. 1-19, IGI Publishing Hershey, PA, USA

[9]   D. Chuda, P. Kratky, J. Tvarozek: Mouse Clicks Can Recognize Web Page Visitors!, Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion) (2015) pp. 21-22

[10]   S. Debnath, N. Ganguly, P. Mitra: Feature Weighting in Content-based Recommendation System using Social Network Analysis. Proceedings of the 17th international conference on World Wide Web (WWW '08) (2008) pp. 1041-1042, ACM, New York, NY, USA

[11]   C. Desrosiers, G. Karypis: A Comprehensive Survey of Neighborhood-based Recommendation Methods. Recommender Systems Handbook, Springer (2011) pp. 107-144

[12]   A. M. Ferman, P. Van Beek, J. H. Errico, M. I. Sezan: Multimedia Content Recommendation Engine with Automatic Inference of User Preferences. Proceedings of the IEEE International Conference on Image Processing (2003) pp. 49-52, IEEE

[13]   B. Ferwerda, M. Schedl, M. Tkalcic: Personality & Emotional States: Understanding Users' Music Listening Needs, User Modeling, Adaptation, and Personalization (2015) Springer International Publishing

[14]   L. Hu, A. Sun, Y. Liu: Your Neighbors Affect your Ratings: on Geographical Neighborhood Influence to Rating Prediction. Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'14) (2014) pp. 345-354

[15]   J. Kiseleva, E. Crestan, R. Brigo, R. Dittel: Modelling and Detecting Changes in User Satisfaction. Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM'14) (2014) pp. 1449-1458, ACM, New York, NY, USA

[16]   M. Kompan, M. Bieliková: Personalized Recommendation for Individual Users Based on the Group Recommendation Principles. Studies in Informatics and Control 22(3) pp. 331-342

[17]   P. Korenek, M. Šimko: Sentiment Analysis on Microblog Utilizing Appraisal Theory. In World Wide Web. 17(4) (2014) pp. 847-867, Springer, USA

[18]   J. Kim, L. Daesung, C. Kyung-Yong: Item Recommendation Based on Context-Aware Model for Personalized u-Healthcare Service. In Multimedia Tools and Applications 71(2) (2014) pp. 855-872

[19]   R. Mukherjee, N. Sajja, S. Sen: A Movie Recommendation System - An Application of Voting Theory in User Modeling. User Modeling and User-Adapted Interaction 13(1-2) (2003) pp. 5-33

[20] M. H. Nadimi-Shahraki B. Mozhode: Cold-start Problem in Collaborative Recommender Systems: Efficient Methods Based on Ask-to-rate Technique. CIT. Journal of Computing and Information Technology 22.2 (2014) pp. 105-113

[21] C. Senot, D. Kostadinov, M. Bouzid, J. Picault, A. Aghasaryan, C. Bernier: Analysis of Strategies for Building Group Profiles. Proceedings of the 18th international conference on User Modeling, Adaptation, and Personalization (UMAP'10) (2010) pp. 40-51, Springer-Verlag, Berlin, Heidelberg

[22] A. Tiroshi, S. Berkovsky, M. Kaafer, D. Vallet, T. Kuflik: Graph-based Recommendations: Make the Most Out of Social Data, In Proceedings of the 22nd international conference on User Modeling, Adaptation, and Personalization (UMAP'14) (2014) pp. 447-458, Springer International Publishing

[23] H. Wang, Y. Liu, P. Yin: Study on User Preferences Modelling Based on Web Mining. International Journal of Information Technology and Management 11, 4 (October 2012) pp. 307-322, Inderscience Publishers

[24] L. Wang, X. Meng, Y. Zhang, Y. Shi: New Approaches to Mood-based Hybrid Collaborative Filtering. Proceedings of the Workshop on Context-Aware Movie Recommendation (CAMRa '10) (2010) pp. 28-33, ACM, New York, NY, USA

[25] W. Wen, L. Chen: Implicit Acquisition of User Personality for Augmenting Movie Recommendations. Proceedings of the 23rd international conference on User Modeling, Adaptation, and Personalization (UMAP'15) (2015) pp. 302-314, Springer International Publishing

[26] Z. Yu, X. Zhou, Y. Hao, J. Gu: TV Program Recommendation for Multiple Viewers Based on user Profile Merging. User Modeling and User-adapted Interaction 16(1) (2006) pp. 63-82